PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 17/14 — A1

(11) International Publication Number: **WO 97/19412**

(43) International Publication Date: 29 May 1997 (29.05.97)

(21) International Application Number: PCT/SE96/00246

(22) International Filing Date: 26 February 1996 (26.02.96)

(30) Priority Data:
PCT/SE95/01371 17 November 1995 (17.11.95) WO
(34) Countries for which the regional or international application was filed: SE et al.

(71) Applicant (for all designated States except US): TERACOM SVENSK RUNDRADIO [SE/SE]; Medborgarplatsen 3, S-118 92 Stockholm (SE).

(72) Inventors; and

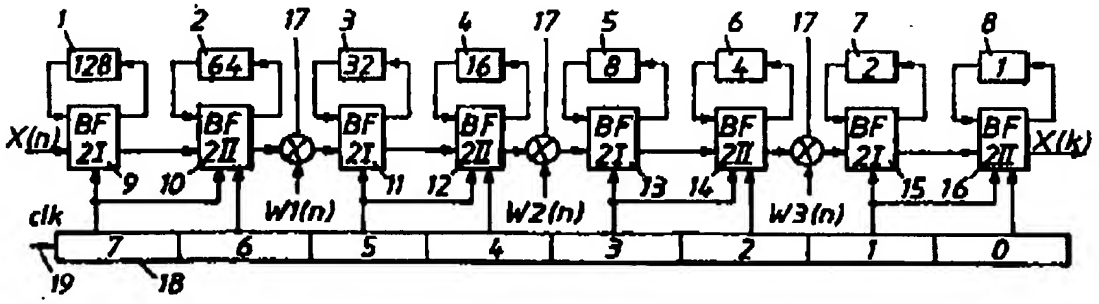
(75) Inventors/Applicants (for US only): HE, Shousheng [SE/SE]; Råbyvägen 15, S-224 57 Lund (SE). TORKELSSON, Mats [SE/SE]; Östen Undéns gata 68, S-227 62 Lund (SE).

(74) Agent: KARLSSON, Berne; Telia Research AB, Rudsjöterrassen 2, S-136 80 Haninge (SE).

(81) Designated States: CA, CN, JP, KR, NO, SG, US, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: IMPROVEMENTS IN OR RELATING TO REAL-TIME PIPELINE FAST FOURIER TRANSFORM PROCESSORS



(57) Abstract

A real-time pipeline processor, which is particularly suited for VLSI implementation, is based on a hardware oriented radix-2^2 algorithm derived by integrating a twiddle factor decomposition technique in a divide and conquer approach. The radix-2^2 algorithm has the same multiplicative complexity as a radix-4 algorithm, but retains the butterfly structure of a radix-2 algorithm. A single-path delay-feedback architecture is used in order to exploit the spatial regularity in the signal flow graph of the algorithm. For a length-N DFT transform, the hardware requirements of the processor proposed by the present invention is minimal on both dominant components: Log_4N-1 complex multipliers, and N-1 complex data memory.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgyzstan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

Improvements in or Relating to Real-Time Pipeline Fast Fourier Transform Processors

The present invention relates to real-time pipeline fast fourier transform processors and, in particular, such processors based on a radix-2^2 algorithm.

Pipeline digital fourier transform processors are a specified class of processors used to perform DFT computations. A real-time pipeline processor is a processor whose processing speed matches the input data rate, i.e. the data acquisition speed for continuous operation. For an FFT processor, this means that a length 'N' DFT must be computed in 'N' clock cycles since the data acquisition speed is one sample per cycle. Pipeline operation enables a partial result, obtained from a preceding stage of the processor, to be immediately used in a following stage, without delay.

FFT processors find application, inter alia, in digital mobile cellular radio systems where there exists considerable constraints on power consumption and chip size. The primary constraining factor may, therefore, be chip complexity, in terms of the number of adders, the number of multipliers, data storage requirements and control complexity, rather than speed of operation.

The present invention emerges from a new approach to the design of real-time pipeline FFT processors. The architecture of a real-time FFT processor, according to the present invention, can be described as a radix-2^2 single-path delay feedback, or R2^2SDF, architecture. Such a processor can operate on the basis of a hardware oriented radix-2^2 algorithm, developed by integrating a twiddle factor decomposition technique in a divide and

conquer approach to form a spatially regular signal flow graph. In a divide and conquer technique the computation of a DFT is decomposed into nested DFTs of shorter length. Divide and conquer techniques are well known in the derivation of fast algorithms and, in the case of the present invention, refer to approachs in which an N-point DFT is decomposed into successively smaller DFTs which are then computed separately and combined to give the final result. The twiddle factor refers to intervening phase shift, or rotational factor. In the present invention, two stages of radix-2 decomposition are performed together and re-decomposed, so that the first stage has only trivial factors which do not require multiplication. However, it should be noted that the two steps are not computed simultaneously.

The algorithm used in the present invention is referred to as a radix-2^2 algorithm because it has the same multiplicative complexity as a radix-4 algorithm but requires radix-2 butterflies in its signal flow graph. The architecture of the processor is described as a single-path delay feedback because only a single data path exists between butterfly stages and each butterfly uses a FIFO buffer in the feedback loop. The signal flow graph is described as spatially regular, because only every alternate column in the SFG has a non-trivial multiplicative operation. This contrasts with an ordinary radix-2 SFG in which there is a non-trivial multiplication in every column in the SFG.

A pipeline DFT processor is characterised by real-time continuous processing of the data sequence passed to the processor. The time complexity of the processor is N and, therefore, it is an AT^2 non-optimal approach with $AT^2 = O(N^2)$, since the area lower bound is O(N). However, in [*"Fourier transform in VLSI"* - *C. D.*

Thompson, IEEE Trans. Comput., C-32(11):1047-1057, Nov. 1983], it has been suggested that for real-time processing a new metric should be introduced, since it is necessarily non-optimal given the time complexity of O(N). Although asymptotically almost all the feasible approaches have reached the area lower bound, [*see S. He and M. Torkelson "A new expandable 2D systolic array for DFT computation based on symbiosis of 1D arrays" Proc. ICA^{3}PP'95, pages 12-19, Brisbane Australia Apr 1995*], one particular class of pipeline processors with the application of recursive Common Factor Algorithm, (collectively known as Fast Fourier Transform), [*see C. S. Burrus "Index mapping for multidimensional formulation of the DFT and convolution" - IEEE Trans. Acoust., Speech, Signal Processing, ASSP-25(3):239-242 June 1977*], has probably the smallest "constant factor" among the approaches that meet the time requirement, due to the least number, O(logN), of processing elements. The difference comes from the fact that an arithmetic unit, especially the multiplier, takes up a much larger area than a digital register in digital VLSI design.

It should be noted that at least Ω(logN) PEs, with multipliers, are needed to meet the real-time processing requirements due to the multiplicative computational complexity of Ω(NlogN) for FFT algorithms. Thus, this is in the nature of a "lower bound" for multiplier requirements. Any optimal architecture for real-time processing will probably have Ω(logN) multipliers.

Another major chip area and energy consumer, for a FFT processor, is the memory requirement for buffering the input data and intermediate results for the computation. For large transforms, this is a dominant factor, [*see E. E. Swartzlander et al. "A radix-4 delay commutator for fast Fourier transform processor implementation" IEEE J. Solid-State Circuits, SC-*

19(5):702-709 Oct 1984: and E. Bidet et al. "A fast single-chip implementation of 8192 complex point FFT" IEEE J. Solid-State Circuits, 30(3):300-305 Mar 1995]. Although there is no formal proof, the area lower bound indicates that the lower bound for the number of registers is likely to be $\Omega(N)$. This is obviously true for any architecture implementing an FFT based algorithm, since the butterfly at the first stage has to take data elements separated by N/r, from the input sequence, where r is a small constant integer, or the radix.

Combining the above arguments suggests a pipeline FFT processor with $\Omega(\log_r N)$ PEs, or multipliers, and $\Omega(N)$ complex word registers. The optimal architecture has to be one that reduces the constant factor, or the absolute number of arithmetic units (multipliers and adders) and memory size, to the minimum.

Some of the known architectures for pipeline processors will now be considered. In order to prevent the comparison between different architectures being affected by sequence order, it will be assumed that the real-time processing task only requires the input time sequence to be in normal order and that the output can be in digit reversed (radix-2, or radix-4) order. This is permissible in such applications as DFT based communication systems, [*see M. Alard and R. Lassalle "Principles of modulation and Channel Coding for digital broadcasting and mobile receivers" EBU Review (224):47-69 Aug 1987*]. DIF type decompositions are used throughout.

The architecture design for pipeline FFT Processors has been the subject of intensive research since the '70s, when a need for real-time processing was identified in such applications as radar signal

processing, [*see L. R. Rabiner and B. Gold "Theory and Application of Digital Signal Processing" - Prentice-Hall 1975*], well before VLSI technology had advanced to the level of system integration. Several architectures have been proposed over the last two decades. These architectures are briefly reviewed below using a unified terminology and functional block diagrams in which the additive butterfly is separated from the multiplier to clearly indicate the hardware requirements. The control and twiddle factor reading mechanism has been omitted for clarity. All data and arithmetic operations are complex and N is a power of 4.

A R2MDC processor is illustrated in Figure 1 of the accompanying drawings, [*see L. R. Rabiner and B. Gold "Theory and Application of Digital Signal Processing" - Prentice-Hall 1975*]. This is a radix-2 Multi-path Delay Commutator architecture which is probably the most straight forward approach to pipeline implementation of a radix-2 FFT algorithm. The input sequence is broken into two parallel data streams flowing forward, with the correct distance between data elements entering the butterfly, scheduled by proper delays. Both butterflies and multipliers are 50% utilised. The processor uses $\log_2 N-2$ multipliers, $\log_2 N$ radix-2 butterflies and $3/2N - 2$ registers (delay elements).

A R2SDF processor is illustrated in Figure 2 of the accompanying drawings, [*E. H. Wold and A. M. Despain "Pipeline and parallel-pipeline FFT processors for VLSI implementation IEEE Trans. Comput. C-35(5):414-426 May 1984*]. This processor uses a radix-2 single path delay-feedback architecture, in which a more efficient use is made of registers, by storing the intermediate butterfly outputs in feedback shift registers. A single data stream passes through the multipliers at every stage. This architecture employs the same number of

butterfly units and multipliers as the R2MDC architecture, but has a much reduced memory requirement, namely N-1 registers. In fact the memory requirement can be described as minimal.

A R4SDF processor is illustrated in Figure 3 of the accompanying drawings, [*see A. M. Despain "Fourier transform computer using CORDIC iterations" IEEE Trans Comput. C-23(10):993-101 Oct 1974*]. This processor uses a radix-4 single path delay feedback architecture and is a radix-4 version of the R2SDF architecture employing CORDIC (Coordinated Rotational Digital Computer) iterations. The multiplier utilisation is increased to 75% because of intermediate storage of 3 out of the 4 radix-4 butterfly outputs. However, the utilisation of the radix-4 butterfly, which is fairly complicated and requires at least 8 complex adders to implement, falls to 25%, [*see J. G. Proakis and D. G. Manolakis "Introduction to Signal Processing" Macmillan 1989*]. A processor implemented in this architecture requires $\log_4 N - 1$ multipliers, $\log_4 N$ full radix-4 butterflies and storage of size N - 1.

A R4MDC processor is illustrated in Figure 4 of the accompanying drawings, [*see L. R. Rabiner and B. Gold "Theory and Application of Digital Signal Processing" - Prentice-Hall 1975*]. This processor uses a radix-4 multi-path delay commutator architecture and is a radix-4 version of the R2MDC architecture. It was used as the architecture for the initial implementation of pipeline FFT processors, [*see E. E. Swartzlander et al. "A radix 4 delay commutator for fast Fourier transform processor implementation" IEEE J. Solid-State Circuits, SC-19(5):702-709 Oct 1984*], and massive wafer scale integration, [*see E. E. Swartzlander et al "A radix 8 wafer scale FFT processor" J. VLSI Signal Processing 4(2,3):165-176 May 1992*]. However, it suffers from a

low, 25%, utilisation of all components. This can only be compensated for in some special applications where four FFTs are being processed simultaneously. Implementation of this processor requires $3\log_4 N$ multipliers, $\log_4 N$ full radix-4 butterflies and $5/2N-4$ registers.

A R4SDC processor is illustrated in Figure 5 of the accompanying drawings. [*see G. Bi and E. V. Jones "A pipelined FFT processor for word-sequential data" IEEE Trans Acoust., Speech, Signal Processing, 37(12):1982-1985 Dec 1989*]. This processor uses a radix-4 single-path delay commutator architecture together with a modified radix-4 algorithm with programmable 1/4 radix-4 butterflies to achieve a higher, 75%, utilisation of multipliers. A combined delay-commutator also reduces the memory requirements, in comparison with the R4MDC architecture, to 2N-2, from 5/2N-1. The butterfly and delay commutator become relatively complicated because of the programmability requirements. The R4SDC architecture has found application in building large single chip pipeline FFT processors for HDTV.

A comparison of the processors described above reveals the distinctive advantages and disadvantages of the different architectures. The delay feedback architectures are always more efficient than the corresponding delay-commutator architectures in terms of memory utilisation, because the stored butterfly outputs can be directly used by the multipliers. Radix-4 algorithm based single-path architectures have a higher multiplier utilisation. However, radix-2 architectures have simpler butterflies which are more efficiently utilised. The present invention is based, at least in part, on these observations.

The present invention is a real-time pipeline

processor which is particularly suited for VLSI implementation. The processor is based on a hardware oriented radix-2^2 algorithm derived by integrating a twiddle factor decomposition technique in a divide and conquer approach. The radix 2^2 algorithm has the same multiplicative complexity as a radix-4 algorithm, but retains the butterfly structure of a radix-2 algorithm. A single-path delay-feedback architecture is used in order to exploit the spatial regularity in the signal flow graph of the algorithm. For a length-N DFT transform, the hardware requirements of the processor proposed by the present invention is minimal on both dominant components: Log_iN-1 complex multipliers, and N-1 complex data memory.

According to a first aspect of the present invention, there is provided a real-time pipeline fast fourier transform processor, characterised in that said processor includes a plurality of paired first and second butterfly means, each of said first butterfly means and each of said second butterfly means having a feedback path between an output therefrom to an input thereto, in that each of said paired butterfly means is linked by a multiplier to an adjacent one of said plurality of paired first and second butterfly means, in that an input data sequence is applied to an input of a first one of said plurality of paired first and second butterfly means, and in that an output data sequence is derived from a last one of said plurality of paired first and second butterfly means.

Preferably, said processor is realised on a VLSI chip.

Preferably, said processor operates on a radix-2^2 algorithm having the same multiplicative complexity as a radix-4 algorithm but employing radix-2 butterflies.

Preferably, only a single data path exists between each butterfly means.

Said first butterfly means may be radix-2 single delay feedback butterflies, and said second butterfly means may be radix-2 single delay feedback butterflies including logic circuitry to implement trivial twiddle factor multiplications.

Said processor may include a synchronisation control means and an address means for twiddle factor reading for each processor stage.

Said first butterfly means may include two adders, two subtractors, and four 2-to-1 multiplexers.

Said second butterfly means may include at least one adder, at least one subtractor, at least two 2-to-1 multiplexers, a 2x2 commutator, and an AND gate with one inverting input and one non-inverting input.

Control signals derived from said synchronisation control means may be applied to the inputs to said AND gate.

Said synchronisation control means and said address means may be implemented as a single binary counter.

A pipeline register may be located between each multiplier and a following butterfly means.

Said processor may include shimming registers for adjusting control signal timing.

Preferably, for a digital fourier transform of length N, said processor includes no more than $\log_4 N - 1$ multipliers, $4\log_4 N$ adders, and a memory size of $N - 1$.

Said processor may be arranged to handle a 256 point FFT, and said processor may have four processing stages in said pipeline, each processing stage separated by a multiplier, each processing stage comprising a first butterfly means with a feedback register, and a second butterfly means with a feedback register.

Said first butterfly means in said first stage may have a one hundred and twenty eight word feedback register, said second butterfly means in said first stage may have a sixty four word feedback register, said first butterfly means in said second stage may have a thirty two word feedback register, said second butterfly means in said second stage may have a sixteen word feedback register, said first butterfly means in said third stage may have an eight word feedback register, said second butterfly means in said third stage may have a four word feedback register, said first butterfly means in said fourth stage may have a two word feedback register and said second butterfly means in said fourth stage may have a one word feedback register.

According to a second aspect of the present incention, there is provided a real-time pipeline fast fourier transform processor, characterised in that said processor operates on a radix-2^2 algorithm having the same multiplicative complexity as a radix-4 algorithm but employing radix-2 butterflies, and in that for a digital fourier transform of length N, said processor includes no more than $\log_4 N - 1$ multipliers, $4\log_4 N$ adders, and a memory size of $N - 1$.

Embodiments of the invention will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 illustrates a known architecture for a

pipeline FFT processor designated as R2MDC.

Figure 2 illustrates a known architecture for a pipeline FFT processor designated as R2SDF.

Figure 3 illustrates a known architecture for a pipeline FFT processor designated as R4SDF.

Figure 4 illustrates a known architecture for a pipeline FFT processor designated as R4MDC.

Figure 5 illustrates a known architecture for a pipeline FFT processor designated as R4SDC.

Figure 6 illustrates a radix-2 butterfly structure, for the present invention, obtained by twiddle factor decomposition.

Figure 7 is a radix-2^2 DIF FFT flow graph for N = 16.

Figure 8 is a radix-2^2 DIF FFT flow graph for N = 64.

Figure 9 illustrates a R2^2SDF pipeline FFT architecture for N = 256, according to the present invention.

Figure 10 illustrates a first butterfly structure used in a R2^2SDF pipeline FFT processor according to the invention.

Figure 11 illustrates a second butterfly structure used in a R2^2SDF pipeline FFT processor according to the invention.

To facilitate an understanding of the present

invention a glossary of the abbreviations used in the specification is set out below:

$\langle . \rangle_N$:	denotes a residue modulo-N operation, e.g. $\langle 7 \rangle_3 = 1$ and $\langle 7 \rangle_4 = 3$
$\Omega(.)$:	lower bound in asymptotic analysis
AT^2:	reference to area-time complexity
CFA:	Common Factor Algorithm
DFT:	Digital, or Discete, Fourier Transform
DIF:	Decimation In Frequency (algorithm) - where a fast algorithm is derived using a divide and conquer approach, if the first step is to divide the input data sequence into a first and second half, equivalent to separating the frequency points by even and odd number, the algorithm is described as a DIF algorithm - in the SFG the frequency points will be in bit-reversed order
DIT:	Decimation In Time (algorithm) - where a fast algorithm is derived using a divide and conquer approach, if the first step is to divide the input data sequence into two, according to its even and odd numbered points, the algorithm is described as a DIT algorithm - in the SFG the input will be in bit reversed order
FIFO:	First In First Out
$R2^2SDF$:	Radix-2^2 Single-path Delay Feedback

FFT: Fast Fourier Transform

Metric: the Hamming distance between two code words - enables a determination of whether, or not, an architecture is optimal to be made

O(.): upper bound in asymptotic analysis

$O(N^2)$: means that the growth rate, for N sufficiently large, is no greater than N^2

PE: Processing Element

SDF: Single-path Delay Feedback

SFG: Signal Flow Graph

VLSI: Very Large Scale Integration

From the observations made in the introduction to this patent specification, it can be seen that a comparison of different architectures for pipeline FFT processors shows that the most desirable hardware oriented algorithm will have the same number of non-trivial multiplications, at the same position in the flowgraph, as a radix-4 algorithm, but will retain the butterfly structure of a radix-2 algorithm. This feature appears in a number of known algorithms. A SFG has been obtained, within a complex "bias" factor, as a result of a constant-rotation/compensation procedure using restricted CORDIC operations, [*see A. M. Despain "Very fast Fourier transform algorithms hardware for implementation" IEEE Trans. Comput. C-28(5):333-341 May 1979*]. Another algorithm combining radix-4 and radix-'4+2', in DIT form, has been used to reduce the scaling noise in a R2MDC architecture, without altering the multiplier requirement, [*see R. Storn "Radix-2 FFT-*

pipeline architecture with reduced noise-to-signal ratio" IEE Proc.-Vis. Image Signal Process. 141(2):81-86 Apr. 1994]. A clear derivation of an algorithm in DIF form directed to the reduction of hardware requirements in the context of pipeline FFT processors has, until now, not been derived.

To avoid confusion with the well known radix-2/4 algorithm and the mixed radix-'4+2' FFT algorithm, [*see E. O. Brigham "The fast Fourier transform and its applications Prentice-Hall 1988*], the algorithm on which the present invention is based is referred to as the radix-2^2 algorithm. This notation clearly reflects the structural relationship of this algorithm to the radix-2 algorithm and the identity between the computational requirements of this algorithm and the radix-4 algorithm.

A DFT of size N is defined by:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \qquad 0 \le k < N \qquad (1)$$

where W_N denotes the Nth primitive root of unity, with its exponent evaluated modulo N. The DFT coefficients are "rotating factors", which are constant length vectors in the complex plane with different phase, or rotation angles. The constant-rotation/compensation procedure proposed by Despain is based on the idea that given a complex bias, all angles can be rotated by successive rotations of a fixed, constant angle. This bias can be compensated at the final stage of the computation. To make the derivation of the new algorithm clearer, consider the first 2 steps in a radix-2 DIF FFT decomposition together. Applying a 3-dimensional linear map,

$$n = \langle \frac{N}{2} n_1 + \frac{N}{4} n_2 + n_3 \rangle N$$
$$k = \langle k_1 + 2k_2 + 4k_3 \rangle N \qquad (2)$$

the CFA algorithm has the form of

$$\begin{aligned} & X(k_1+2k_2+4k_3) \\ &= \sum_{n_3=0}^{\frac{N}{4}-1} \sum_{n_2=0}^{1} \sum_{n_1=0}^{1} x(\frac{N}{2}n_1+\frac{N}{4}n_2+n_3)\, W_N^{(\frac{N}{2}n_1+\frac{N}{4}n_2+n_3)(k_1+2k_2+4k_3)} \\ &= \sum_{n_3=0}^{\frac{N}{4}-1} \sum_{n_2=0}^{\cdot} \{[x(\frac{N}{4}n_2+n_3)+(-1)^{k_1} x(\frac{N}{4}n_2+n_3+\frac{N}{2}]\, W_N^{(\frac{N}{4}n_2+n_3)k_1}\}T \\ & \qquad \text{where} \quad T = W_N^{(\frac{N}{4}n_2+n_3)(2k_2+4k_3)} \end{aligned} \qquad (3)$$

If the expression within the braces is computed with a butterfly structure before further decomposition, an ordinary radix-2 DIF FFT will be obtained. The key concept behind the new algorithm is to extend the second stage of decomposition to the remaining DFT coefficients, including the twiddle factor

$$W_N^{(\frac{N}{4}n_2+n_3)k_1}$$

to exploit the exceptional values in multiplication before the butterfly is constructed. Decompose the composite twiddle factor observing that:-

$$\begin{aligned} & W_N^{(\frac{N}{4}n_2+n_3)(k_1+2k_2+4k_3)} \\ &= W_N^{Nn_2k_3} W_N^{\frac{N}{4}n_2(k_1+2k_2)} W_N^{n_3(k_1+2k_2)} W_N^{4n_3k_3} \\ &= (-j)^{n_2(k_1+2k_2)} W_N^{n_3(k_1+2k_2)} W_N^{4n_3k_3} \end{aligned} \qquad (4)$$

Substituting equation (4) into equation (3) and expanding the summation with index n_1, yields, after

simplification, a set of four DFTs of length N/4,

$$X(k_1+2k_2+4k_3)=\sum_{n_3=0}^{\frac{N}{4}-1}[H(k_1,k_2,n_3)\,W_N^{n_3(k_1+2k_2)}]\,W_{\frac{N}{4}}^{n_3k_3} \qquad (5)$$

where

$$H(k_1,k_2,n_3)=[x(n_3)+(-1)^{k_1}x(n_3+\frac{N}{2})] + (-j)^{(k_1+2k_2)}[x(n_3+\frac{N}{4})+(-1)^{k_1}x(n_3+\frac{3}{4}N)] \qquad (6)$$

The expressions in [ ] on the right hand side of equation (6) represent a first butterfly BF I, the entire expression on the right hand side of equation (6) represents a second butterfly BF II. Equation (6) represents the first two stages of butterflies with only trivial multiplication in the flow graph, as BF I and BF II, in Figure 6. After these two stages, full multipliers are required to compute the multiplications by the decomposed twiddle factor:

$$W_N^{n_3(k_1+2k_3)}$$

in equation 6, as shown in Figure 6. It should be noted that the order of the twiddle factors is different from that of a radix-4 algorithm.

Applying this CFA procedure recursively to the remaining DFTs of length N/4 in equation (5), yields the complete radix-2^2 DIF FFT algorithm. An N = 16 example is shown in figure 7 and an N = 64 example is shown in Figure 8. In Figure 8 the small diamonds represent trivial multiplication by:

$$W_N^{\frac{N}{4}} = -j$$

which involves only real-imaginary swapping and sign inversion.

The radix-2^2 algorithm has the same multiplicative complexity as the radix-4 algorithm, but retains the radix-2 butterfly structure. The multiplicative operations are so arranged that only every other stage has non-trivial multiplications. This is a substantial structural advantage over other algorithms when pipeline/cascade FFT architectures are considered.

By applying the radix-2^2 DIF FFT algorithm, derived above, to a R2SDF architecture, the new and efficient radix-2^2SDF architecture of the present invention is obtained. This architecture requires a minimum of hardware resource, compared with the known architectures discussed in the introduction to this specification, because of the reduced multiplicative complexity and the preservation of spatial regularity for both additive and multiplicative operations in the SFG, as shown in Figures 7 and 8.

Figure 9 illustrates the architecture of a real-time pipeline FFT processor, according to the present invention, for N=256. The similarity between the data path in this processor and the R2SDF architecture and the reduced number of multipliers should be noted.

Referring now to Figure 9, the input data sequence is passed to the first, 9, of a pair butterfly units, 9 and 10. A one hundred and twenty eight word feedback register, 1, links the output of butterfly 9, to its input. The second butterfly unit 10 has a sixty four word feedback register 2. Multiplier 17 links the first

stage of the processor, comprising butterfly units 9 and 10 to the second stage of the processor comprising butterfly units 11 and 12, and multiplies the data stream by the twiddle factor W1(n). It should be noted at this point that the structure of butterfly units 9, 11, 13 and 15, differs from butterfly units 10, 12, 14, and 16, see below. Butterfly units 11 and 12 are provided with feedback registers 3 and 4 having a thirty two word and sixteen word capacity respectively. A multiplier 17, located between the second and third stage of the processor, multiplies the data stream by twiddle factor W2(n). The third stage of the processor comprises butterflies 13 and 14, together with eight word feedback register 5 and four word feedback register 6. Again a multiplier 17, located between the third and fourth stages, of the processor multiplies the data stream by twiddle factor W3(n). The fourth stage of the processor comprises butterfly units 15 and 16 together with two word feedback register 7 and one word feedback register 8. The output sequence, X(k) is derived from the output of the fourth stage of the processor. The binary counter 18, is clocked by a clock signal 19. The binary counter 18 acts as a synchronisation controller and address counter for the twiddle factors used between each stage of the processor.

The two types of butterfly used in the processor are BF2I and BF2II. The BF2I butterfly is similar to the R2SDF butterfly. The BF2II butterfly contains the logic needed to implement the trivial twiddle factor multiplications. Because of the spatial regularity of the radix-2^2 algorithm, the synchronisation control of the processor is particularly simple and is, as described above, implemented with a ($\log_2 N$)-bit counter.

The type BF2I butterfly is illustrated in Figure 10. The butterfly unit comprises two adders, 21, two

subtractors, 22, and four multiplexers 23, connected as shown in Figure 10. Operation of the multiplexers is controlled by control signal 27, as will be explained later.

The type BF2II butterfly is illustrated in Figure 11. It is similar in construction to the type BF2I butterfly, but with the addition of a 2x2 commutator, 26, and a logic gate, 24. The logic gate 24 is an AND gate with one inverted input. Control signal 25 is applied to the inverted input of AND gate 24, and control signal, 27, which is also applied to the multiplexers 23, is applied to the non-inverted input of AND gate 24. The output from AND gate 24 drives commutator 26.

The scheduled operation of the R2^{2}SDF processor is as follows. On the first N/2 cycles, the 2-to-1 multiplexers, 23, in the first butterfly module switch to position "0", and the butterfly is idle. The input data from the left is directed to the shift registers until they are filled. On the next N/2 cycles, the multiplexers, 23, turn to position "1", the butterfly unit computes a 2-point DFT with the incoming data and the data stored in the shift registers.

$$\begin{aligned} Z1(n) &= x(n) + x(n+N/2) \\ Z1(n+N/2) &= x(n) - x(n+N/2) \end{aligned} \qquad 0 \le n < N/2 \qquad (7)$$

The butterfly output Z1(n) is sent to apply the twiddle factor and Z1(n+N/2) is sent back to the shift registers to be "multiplied" in the next N/2 cycles when the first half of the next frame of the time sequence is loaded. The operation of the second butterfly is similar to that of the first one, except the "distance" of the butterfly input sequence is just N/4 and the

trivial twiddle factor multiplication is implemented by real-imaginary swapping by commutator 26 and controlled add/subtract operations. This requires a two bit control signal, 25 and 27, from the synchronising counter, 18. The data then passes through a full complex multiplier, 17, working at 75% utility, to produce the results of the first level of radix-4 DFT word by word. Further processing repeats this pattern with the distance of the input data decreasing by half at each consecutive butterfly stage. After N-1 clock cycles, the complete DFT transform result streams out to the right of processor, see Figure 9, in bit-reversed order. The next frame of the transform can then be processed without pausing, because of the pipelined processing at each stage of the processor.

In a practical implementation of the radix-2^2SDF processor, pipeline registers should be inserted between each multiplier and butterfly stage to improve performance. Shimming registers are also needed so that the control signals comply with the revised timing. The latency of the output is then increased to $N-1+3(\log_4 N-1)$ without affecting the throughput rate.

The hardware requirements of radix-2^2SDF processor architecture, as compared with known architectures, is set out in the table below, which lists the number of complex multipliers, adders, memory size and control complexity.

	multiplier #	adder #	memory size	control
R2MDC	$2(\log_4 N-1)$	$4\log_4 N$	$3N/2-2$	simple
R2SDF	$2(\log_4 N-1)$	$4\log_4 N$	$N-1$	simple
R4SDF	$\log_4 N-1$	$8\log_4 N$	$N-1$	medium

R4MDC	$3(\log_4 N-1)$	$8\log_4 N$	$5N/2-4$	simple
R4SDC	$\log_4 N-1$	$3\log_4 N$	$2N-2$	complex
$R2^2SDF$	$\log_4 N-1$	$4\log_4 N$	$N-1$	simple

The table shows that the $R2^2SDF$ architecture has reached the minimum requirement for both multipliers and storage requirements, and is second, with regard to the number of adders, to only the R4SDC architecture. This means that the $R2^2SDF$ architecture is ideal for the VLSI implementation of pipeline FFT processors.

CLAIMS

1. A real-time pipeline fast fourier transform processor, characterised in that said processor includes a plurality of paired first and second butterfly means, each of said first butterfly means and each of said second butterfly means having a feedback path between an output therefrom to an input thereto, in that each of said paired butterfly means is linked by a multiplier to an adjacent one of said plurality of paired first and second butterfly means, in that an input data sequence is applied to an input of a first one of said plurality of paired first and second butterfly means, and in that an output data sequence is derived from a last one of said plurality of paired first and second butterfly means.

2. A real-time pipeline fast fourier transform processor as claimed in claim 1, characterised in that said processor is realised on a VLSI chip.

3. A real-time pipeline fast fourier transform processor as claimed in either claim 1, or claim 2, characterised in that said processor operates on a radix-2^2 algorithm having the same multiplicative complexity as a radix-4 algorithm but employing radix-2 butterflies.

4. A real-time pipeline fast fourier transform processor as claimed in any previous claim, characterised in that only a single data path exists between each butterfly means.

5. A real-time pipeline fast fourier transform processor as claimed in any previous claim, characterised in that said first butterfly means are

radix-2 single delay feedback butterflies, and in that said second butterfly means are radix-2 single delay feedback butterflies including logic circuitry to implement trivial twiddle factor multiplications.

6. A real-time pipeline fast fourier transform processor as claimed in any previous claim, characterised in that said processor includes a synchronisation control means and an address means for twiddle factor reading for each processor stage.

7. A real-time pipeline fast fourier transform processor as claimed in either claim 5 or 6, characterised in that said first butterfly means includes two adders, two subtractors, and four 2-to-1 multiplexers.

8. A real-time pipeline fast fourier transform processor as claimed in any of claims 5 to 7, characterised in that said second butterfly means includes at least one adder, at least one subtractor, at least two 2-to-1 multiplexers, a 2x2 commutator, and an AND gate with one inverting input and one non-inverting input.

9. A real-time pipeline fast fourier transform processor as claimed in claim 8, when appended to claim 6, or 7, characterised in that control signals derived from said synchronisation control means are applied to the inputs to said AND gate.

10. A real-time pipeline fast fourier transform processor as claimed in any of claims 6 to 9, characterised in that said synchronisation control means and said address means are implemented as a single binary counter.

11. A real-time pipeline fast fourier transform processor as claimed in any previous claim, characterised in that a pipeline register is located between each multiplier and a following butterfly means.

12. A real-time pipeline fast fourier transform processor as claimed in claim 11, characterised in that said processor includes shimming registers for adjusting control signal timing.

13. A real-time pipeline fast fourier transform processor as claimed in any previous claim, characterised in that for a digital fourier transform of length N, said processor includes no more than $\log_t N - 1$ multipliers, $4\log_t N$ adders, and a memory size of N - 1.

14. A real-time pipeline fast fourier transform processor as claimed in any previous claim, characterised in that said processor is arranged to handle a 256 point FFT, in that said processor has four processing stages in said pipeline, each processing stage being separated by a multiplier, and in that each processing stage comprises a first butterfly means with a feedback register, and a second butterfly means with a feedback register.

15. A real-time pipeline fast fourier transform processor as claimed in claim 14, characterised in that said first butterfly means in said first stage has a one hundred and twenty eight word feedback register, said second butterfly means in said first stage has a sixty four word feedback register, said first butterfly means in said second stage has a thirty two word feedback register, said second butterfly means in said second stage has a sixteen word feedback register, said first butterfly means in said third stage has an eight word feedback register, said second butterfly means in said

third stage has a four word feedback register, said first butterfly means in said fourth stage has a two word feedback register and said second butterfly means in said fourth stage has a one word feedback register.

16. A real-time pipeline fast fourier transform processor, characterised in that said processor operates on a radix-2^2 algorithm having the same multiplicative complexity as a radix-4 algorithm but employing radix-2 butterflies, and in that for a digital fourier transform of length N, said processor includes no more than $\log_4 N - 1$ multipliers, $4\log_4 N$ adders, and a memory size of N - 1.

WO 97/19412 PCT/SE96/00246

Fig. 1

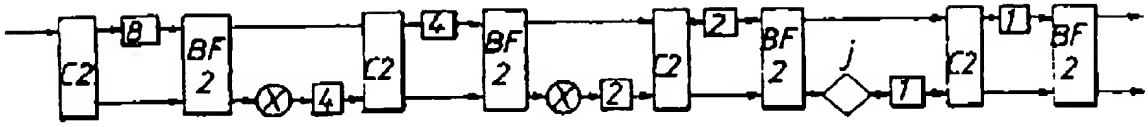


Fig. 2

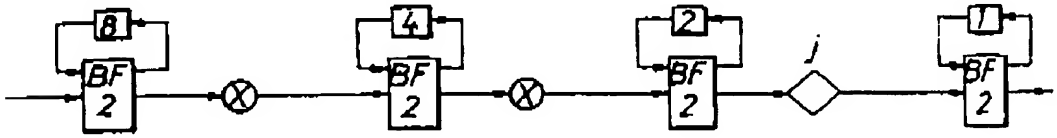


Fig. 3

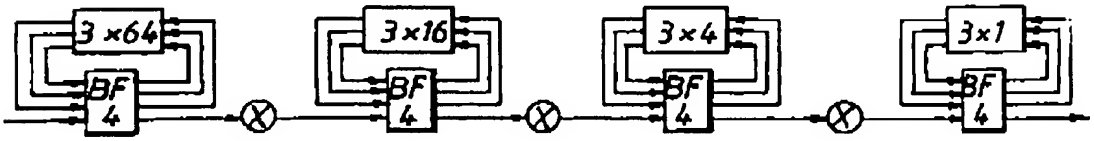


Fig. 4

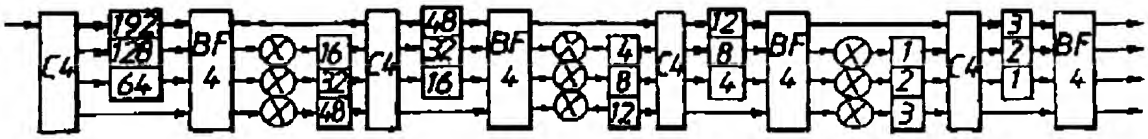


Fig. 5

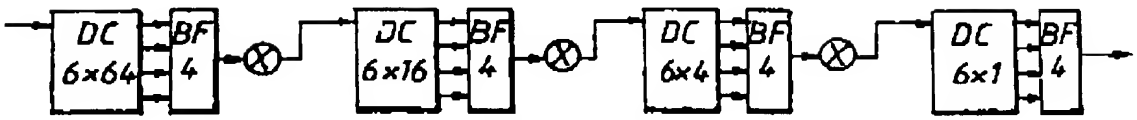


Fig. 6

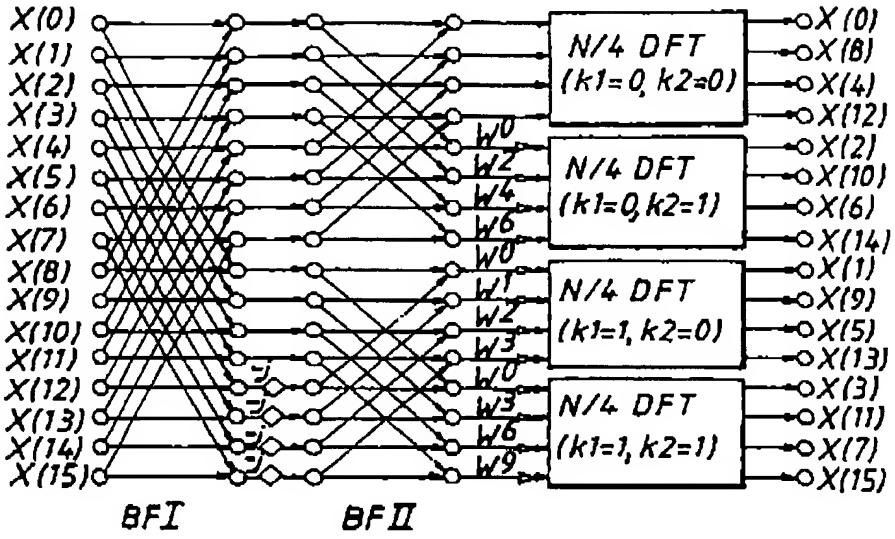
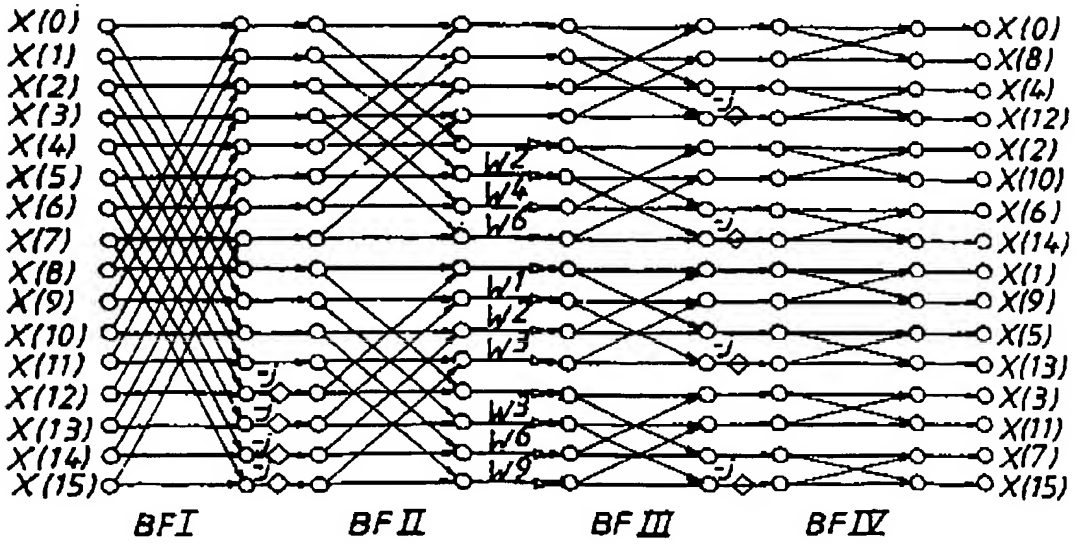

X(0)
X(1)
X(2)
X(3)
X(4)
X(5)
X(6)
X(7)
X(8)
X(9)
X(10)
X(11)
X(12)
X(13)
X(14)
X(15)
N/4 DFT (k1=0, k2=0)
N/4 DFT (k1=0, k2=1)
N/4 DFT (k1=1, k2=0)
N/4 DFT (k1=1, k2=1)
W0
W2
W4
W6
W0
W1
W2
W3
W0
W3
W6
W9
-j
X(0)
X(8)
X(4)
X(12)
X(2)
X(10)
X(6)
X(14)
X(1)
X(9)
X(5)
X(13)
X(3)
X(11)
X(7)
X(15)
BFI
BFII

Fig. 7


X(0)
X(1)
X(2)
X(3)
X(4)
X(5)
X(6)
X(7)
X(8)
X(9)
X(10)
X(11)
X(12)
X(13)
X(14)
X(15)
W2
W4
W6
W1
W2
W3
W3
W6
W9
-j
X(0)
X(8)
X(4)
X(12)
X(2)
X(10)
X(6)
X(14)
X(1)
X(9)
X(5)
X(13)
X(3)
X(11)
X(7)
X(15)
BFI
BFII
BFIII
BFIV

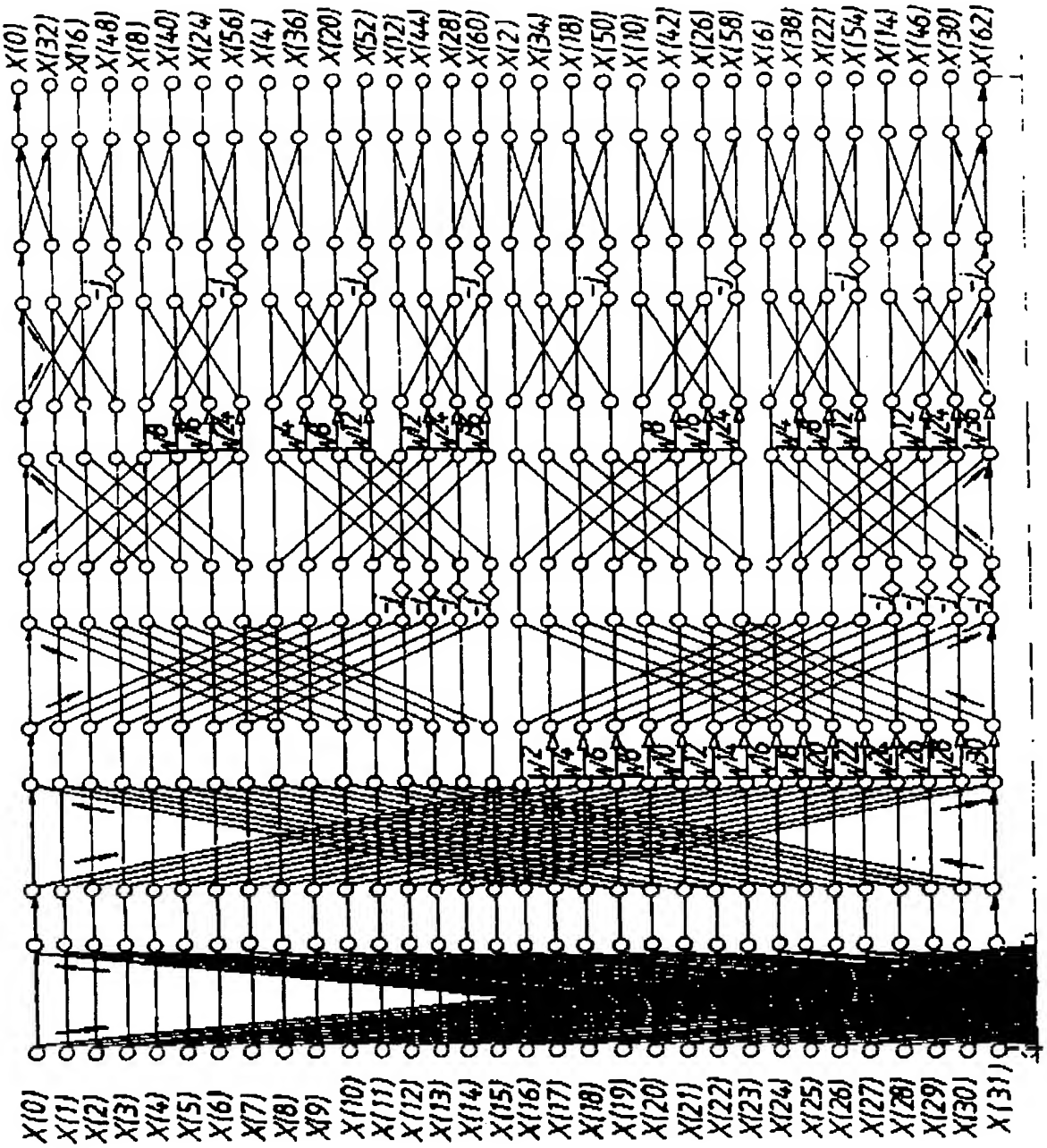

X(0)
X(32)
X(16)
X(48)
X(8)
X(40)
X(24)
X(56)
X(4)
X(36)
X(20)
X(52)
X(12)
X(44)
X(28)
X(60)
X(2)
X(34)
X(18)
X(50)
X(10)
X(42)
X(26)
X(58)
X(6)
X(38)
X(22)
X(54)
X(14)
X(46)
X(30)
X(62)
X(0)
X(1)
X(2)
X(3)
X(4)
X(5)
X(6)
X(7)
X(8)
X(9)
X(10)
X(11)
X(12)
X(13)
X(14)
X(15)
X(16)
X(17)
X(18)
X(19)
X(20)
X(21)
X(22)
X(23)
X(24)
X(25)
X(26)
X(27)
X(28)
X(29)
X(30)
X(31)

Fig. 8

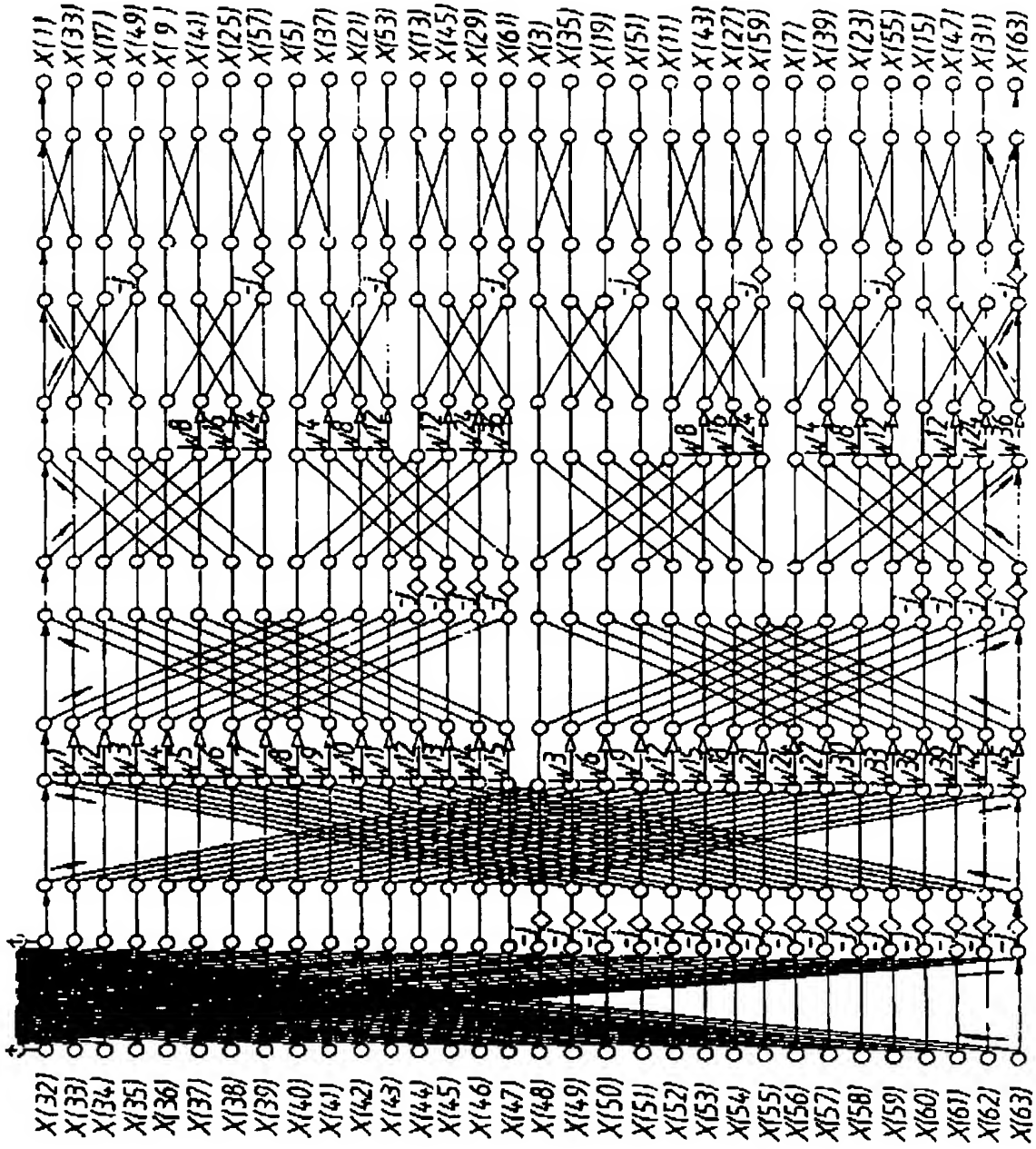

X(1) X(33) X(17) X(49) X(9) X(41) X(25) X(57) X(5) X(37) X(21) X(53) X(13) X(45) X(29) X(61) X(3) X(35) X(19) X(51) X(11) X(43) X(27) X(59) X(7) X(39) X(23) X(55) X(15) X(47) X(31) X(63)
X(32) X(33) X(34) X(35) X(36) X(37) X(38) X(39) X(40) X(41) X(42) X(43) X(44) X(45) X(46) X(47) X(48) X(49) X(50) X(51) X(52) X(53) X(54) X(55) X(56) X(57) X(58) X(59) X(60) X(61) X(62) X(63)

Fig. 8

WO 97/19412 PCT/SE96/00246

Fig. 9

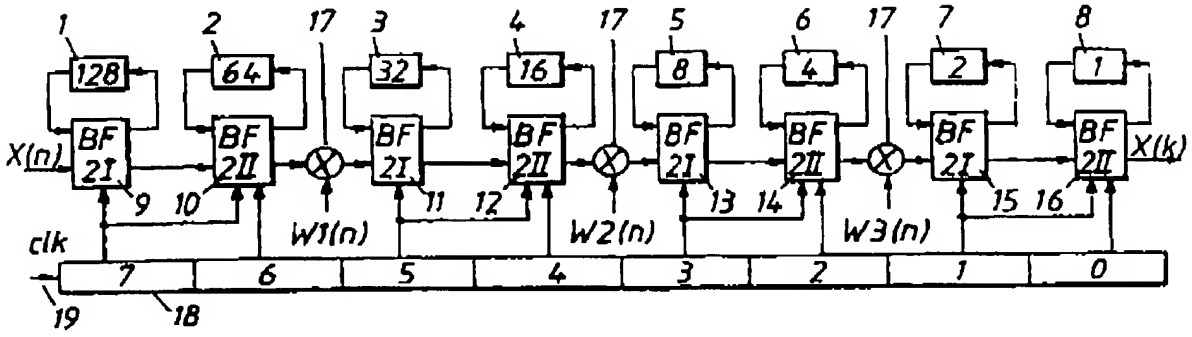


Fig. 10

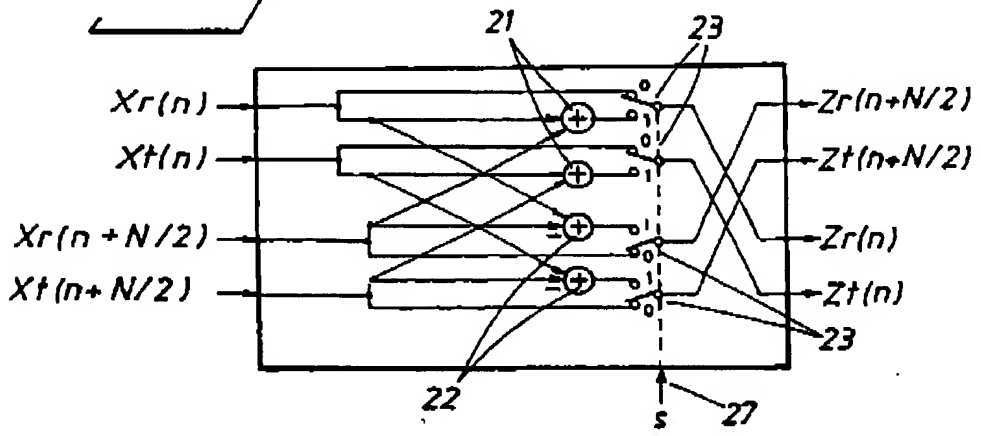
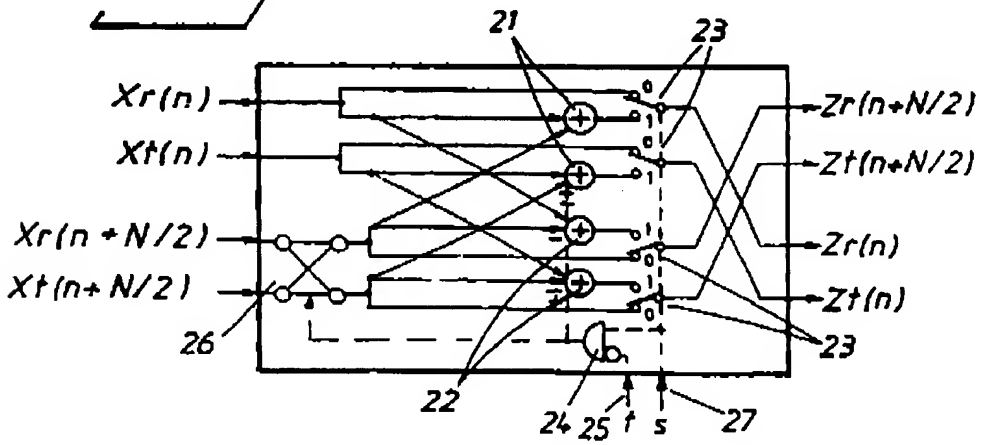


Fig. 11



INTERNATIONAL SEARCH REPORT

International application No.
PCT/SE 96/00246

A. CLASSIFICATION OF SUBJECT MATTER

IPC6: G06F 17/14
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC6: G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPODOC, WPI

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 4563750 A (CLARKE), 7 January 1986 (07.01.86), figure 4 --	1-16
A	US 5293330 A (SAYEGH), 8 March 1994 (08.03.94), abstract --	16

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" erlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 May 1997	07.05.97
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Jan Silfverling Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/SE 96/00246

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 4563750 A	07/01/86	NONE	
US 5293330 A	08/03/94	NONE	